

## RESEARCH

# Semi-supervised Learning for the BioNLP Gene Regulation Network

Thomas Provoost\* and Marie-Francine Moens

\*Correspondence:

thomas.provoost@cs.kuleuven.be  
Computer Science Department,  
KU Leuven, Celestijnenlaan 200A,  
3001 Heverlee, BE  
Full list of author information is  
available at the end of the article

## Abstract

**Background:** The BioNLP Gene Regulation Task has attracted a diverse collection of submissions showcasing state-of-the-art systems. However, a principal challenge remains in obtaining a significant amount of recall. We argue that this is nonetheless an important quality for Information Extraction tasks in this field. We propose a semi-supervised framework, leveraging a wide corpus of unannotated data available to us. In this framework, the annotated data is used to find plausible candidates for positive data points, which are included in the machine learning process. As this is a method principally designed for gaining recall, we further explore additional methods to improve precision on top of this. These are: differential regularisation in the SVM framework, and filtering out unlabelled examples based on a probabilistic rule-finding method. The latter method also allows us to add candidates for negatives from unlabelled data, a method not viable in the unfiltered approach.

**Results:** We replicate one of the original participant systems, and modify it to incorporate our methods. This allows us to test the extent of our proposed methods by applying them to the GRN task data. We find a considerable improvement in recall compared to the baseline system. We also investigate the evaluation metrics and find several mechanisms explaining a bias towards precision. Furthermore, these findings uncover an intricate precision-recall interaction, depriving recall of its habitual immediacy seen in traditional machine learning setups.

**Conclusion:** Our contributions are twofold:

- 1 An exploration of a novel semi-supervised pipeline. We have succeeded in employing additional knowledge through adding unannotated data points, while responding to the inherent noise of this method by imposing an automated, rule-based pre-selection step.
- 2 A thorough analysis of the evaluation procedure in the Gene Regulation Shared Task. We have performed an in depth inquiry of the Slot Error Rate, responding to arguments that lead to some design choices of this task. We have furthermore uncovered complexities in the interplay of precision and recall that negate the customary behaviour commonplace to the machine learning engineer.

**Keywords:** Machine Learning; Relation Learning; Semi-supervised

## Introduction

The set of BioNLP shared tasks [1] form a biannual challenge used by many to apply and develop state-of-the-art methods in the field of biomedical information extraction (IE). In 2013 in its third instalment, it again succeeded in attracting a considerable amount of contributions from an international community of re-

searchers. This work is spread over six different subtasks, each with a focus on fine-grained IE to construct knowledge bases in their respective domain.

The Gene Regulation Network subtask [2] tries to attain the construction of a relation network encompassing the extracted knowledge, in order to build models to represent the behaviour of a system. This network can then serve as a base representing current knowledge, and be leveraged for making inferences and predictions, i.e. towards experiment design. In the case of this particular task, this system entails the whole of molecular interactions between genes and proteins in a specific bacterium, the *bacillus subtilis*. Participants are asked to extract a regulation network from sentences taken from PubMed abstracts describing these phenomena. This network is comprised of six different types of relations, which are related into a small hierarchy (see Table 3).

When looking at the results that were obtained, it is apparent that there is substantial room for improvement. While suitable precision is showcased by several of the systems, no team managed to obtain a high recall. Almost all submission were very conservative in their predictions; consequently only an average recall of 26.6% was obtained. Nevertheless, covering a sufficient proportion of true positives is fundamental in many practical applications, such as knowledge base construction or hypothesis generation. Indeed, the interest in developing systems for inference and/or prediction lies all the more in the retrieval of a sizeable hypothesis set, rather than reaching only those that can be found with high confidence. In this paper we therefore wish to focus on this goal of covering sufficient true positives. We further investigate possible causes of the low recall seen in all participant systems, and explore some techniques to counterbalance this conservative nature.

We start by proposing the exploitation of additional unannotated data to enhance an existing model used for this task. By working in a semi-supervised fashion, a learner can be made more aware of the wide variety of patterns encoding a relationship. This happens at the cost of introducing more noise, since there is no reliable way of labelling this extra data. Basing ourselves on the model of [3], that achieved a second place for this task, we explore how semi-supervised techniques can improve the performance that this system obtains in its supervised form. We further investigate several techniques to counterbalance the added noise by these methods. Next to the traditional measure of applying differential regulation parameters, we go on to develop a novel method based on probabilistic rule-finding.

Next, we look at the experimental set-up and compare the results of the proposed methods. We also discuss some of the properties of this task, and evaluate how these can impact performance in terms of precision and recall. This influence can be both direct, e.g. because of data skewness or pre-scoring processing, and indirect. An example of the latter is found in the choice of the final scoring metric (the Slot Error Rate), altering some of the parameter choices when designing and selecting a model.

The section thereafter reviews related work. We finish with conclusions and future research questions.

## Methodology

### Baseline model

We base ourselves on the model of [3]. The main reasons for this are as follows:

- Their model came in second place, showing decent performance;
- Unlike the winning entry, their model does not use hand-crafted rules, and is based on Support Vector Machines. Their set-up therefore lends itself perfectly to replication, and extension into a semi-supervised framework as described below.

The main configuration of the system of [3] is a collection of Support Vector Machines (SVMs, see [4]), one per relation type. The authors construct a data point for each couple of genic entities in a sentence, effectively considering all potential agent/target pairs for the relations. The kernel used is a Gaussian RBF kernel (see [5] for the seminal work, and [6] for a good overview).

The novelty of [3] lies in the feature construction. The feature vectors are a concatenation of *local* features  $f_{base}$ , complemented by what is referred to as *context* features,  $f_{context}$ . The local features consist mostly of vocabulary and part-of-speech information for the words that the entities comprise, with different parts for the agent and target entities. The context part is then constructed in the following fashion, also separately for both entities:

$$f_{context}(\omega) = \frac{1}{Z} \sum_{\omega_i \in sentence} \alpha^{d(\omega, \omega_i)} f_{base}(\omega_i)$$

with  $\omega$  being the words of the entities at hand, and the sum going over all the  $\omega_i$  words in the sentence.  $d(\omega, \omega_i)$  is the distance in number of words between  $\omega$  and  $\omega_i$ . This is in essence an average of the vectors encoding the different non-entity words in the sentence, weighted inversely by their distance to the entity words.  $\alpha$  is a constant controlling how fast the weights decay with distance, and  $Z$  is a normalisation factor. We direct the reader to the work of [3] for further details.

A few specific differences are to be noted between our implementation and that of the submitted system. We use the LibSVM [7] package as provided by the Scikit-learn Toolbox [8]; this difference in library used should be of minor influence on results, and we are indeed able to replicate their performance.

#### A semi-supervised method inspired by Multiple-Instance Learning (MIL)

The main issue of a fully supervised system is the difficulty to generalise towards unseen patterns. This problem is more apparent the sparser the data, and the richer the representation. With our baseline system having an elaborate feature representation, we suspect this to be a big factor in this framework. Furthermore, new data points will likely entail unseen words, in part counterbalancing the effectiveness of this sort of feature scheme, albeit widely used in NLP situations (as shown in e.g. [9] and [10]). Because of these reasons, the base system is likely to suffer from a poor generalisability, as also testified by its poor recall score.

A corpus of related, but unseen data points can provide a source of new patterns to incorporate in our learner. Of course, the main obstacle is the lack of labelling for this data; we have no knowledge what points are to be marked as positive. Instrumental in any semi-supervised framework are therefore:

- An approximation method to identify the labelling of unseen data. As this can never fully substitute the precision of supplied annotations, the uncertainty in this introduces additional noise. Hence also the need for the next item:

- Means of managing the uncertainty in adding unlabelled data. Since the labellings now contain more noise, this inherently changes the optimal learning strategy; a semi-supervised method needs to take this into account.

We propose a framework that is inspired by Multiple-instance Learning (see [11] for a good overview). In this line of methods, the classifier is trained on a set of ‘bags’ of data points, with the defining property that positive bags are only known to be partly containing positively labelled points. The negative bags on the other hand are certain to effectively contain only negative points. As shown in [12], one use case for this setup is exactly relation learning, in the event of having a set of known relations between two entities, but when no sentence-level annotations are available.

Contrary to this framework, we do dispose of fine-grained annotations in our labelled data set. However, the structure of these MIL problems points us to the aforementioned approximation method to add unlabelled data to the training data. Namely, the following observation is used: if a biological relation exists between two entities (as seen in the labelled data), there is a substantial probability that another (unlabelled) sentence containing both entities will also encode this relation. We therefore add any data point from the unannotated corpus that is composed of two such entities to the training set, labelling it positive. Note that, since our main goal is to introduce new patterns to the classifier, we also use the vocabulary from these sentences when constructing feature vectors. This ensures that we use an unbiased representation of these data points.

Opposite to the case of positive examples, the same inference can not be performed here for negative data points. Absence from a small set of known relations only marginally changes the probabilities on these points. We therefore refrain from adding negatives from the unlabelled data, barring further methods to obtain a more accurate selection. As will be seen, the pre-selection filter we develop in the following subsection will allow for exactly this; there we will revisit this choice.

We will refer to the above method as the ‘basic’ method (cfr. in results Table 2 the entry [BASIC]), as opposed to the systems augmented with the techniques described below.

### Methods of counterbalancing the added noise

Whenever reliability of labelling is affected, this directly influences precision. The basic method proposed above is guaranteed to introduce new patterns to the classifier, which is expected to improve recall. However, this comes at the cost of adding uncertainty to the labelling of the data, which is prone to an increase in false positives.

In this part, we will look at different methods to counter this effect and maintain adequate precision. We study the effects of a general method known to deal with different kinds of noise, namely having a non-constant regularisation parameter in the SVM. We then move on to develop a method of pre-selecting the data that is added from the unlabelled corpus, leading to a more fine-grained control of the introduced uncertainty.

### *Differential regularisation*

A conventional way in MIL to deal with noisy training examples comes with the observation that, in the traditional setup, only the positive data points are plagued by this noise. Hence, in a soft-margin SVM framework (as developed by [13]), a different regularisation policy is introduced for positive and negative examples, as first proposed by [14]. Let  $\chi^+, \chi^-$  be the set of positive and negative data points respectively, and  $\phi(x)$  be the feature representation for  $x$ , this then leads to the following optimisation formulation:

$$\min_{w, b, \xi_x} \left( \frac{1}{2} \|w\|^2 + \mathcal{C}^+ \sum_{x \in \chi^+} \xi_x + \mathcal{C}^- \sum_{x \in \chi^-} \xi_x \right)$$

subject to:

$$\begin{aligned} \langle w, \phi(x) \rangle + b &\geq 1 - \xi_x, & \forall x \in \chi^+ \\ \langle w, \phi(x) \rangle + b &\leq -1 + \xi_x, & \forall x \in \chi^- \\ \xi_x &\geq 0, & \forall x \in \chi^+ \cup \chi^- \end{aligned}$$

$w$  is the weight vector that defines the separating hyperplane together with the constant  $b$  as a bias term. The  $\xi_x$  serve in this optimisation problem as *slack variables*, allowing a trade-off of maximising the margin against having a few points surpassing that margin. By having two regularisation constants  $\mathcal{C}^+$  and  $\mathcal{C}^-$ , we can allow the margin for positive points to be ‘softer’, accounting for the additional uncertainty in this subset.

### *An automatic rule-detection algorithm for pre-selection of unannotated data*

Many machine learning systems that serve a specific application make use of a framework that incorporates specialist knowledge. A prevalent mechanism for this is by having some rule-based pre-/post-processing. We propose a method for extracting some of this knowledge from the labelled data in an automated fashion. In the framework of our semi-supervised system, this can then be used to obtain a more fine-grained selection from our unlabelled corpus.

As we are dealing with a pre-selection step on what is expected to be positive, our main focus is on detecting sufficient conditions in the feature space for negativity. In order to find such a rule implicitly present in the data, we observe the following:

$$\begin{aligned} (f_i \in V_i \rightarrow 0) &\implies (1 \rightarrow f_i \notin V_i) \\ &\implies P(f_i \in V_i \mid 1) = 0 \end{aligned}$$

where  $f_i$  is the  $i$ th feature of a data point,  $V_i$  a set of values, and 0, 1 have been used as shorthand for the (negative resp. positive) labelling of that point. The extension towards rules that conjoin several features is immediate.

While the above observation is sufficient for a negative labelling, it is by no means necessary, i.e. finding a zero frequency can not exclude chance, especially in small datasets. To see how much of a factor  $f_i$  effectively is in the labelling of the point,

one could look at probabilistic measures such as Mutual Information, Bayes Factor or the Kullback-Leibler divergence. However, most of these measures are only meaningful on non-zero probabilities, mainly because of the occurrence of logarithms or divisions of these probabilities.

To escape the ill-behaved nature in this situation, we look at the probability mass  $P(f_i \in V_i | 0)$ , and demand it to be above a certain threshold. This avoids the confusion of rarely occurring feature values with rules, since this significantly lowers the probability that all mass ends up with negative points by chance.

In the algorithm we construct below, we select good features to extract rules from, as well as combinations of two feature dimensions. While it is feasible to explore the use of even more features simultaneously in a rule, we abstain from doing so to preserve the balance between exhaustiveness and system performance. The steps to efficiently find these rules are as follows:

```

1: initialise  $R = [], T = []$ 
2: for all  $i$  do divide the values for  $f_i$  into two bins  $V_i, \overline{V_i}$ 
3: end for
4: for all  $i$  do
5:   if  $\text{Count}(f_i \in V_i, 0) > \text{threshold}$  then
6:     Add  $i$  to  $T$ 
7:   if  $\text{Count}(f_i \in V_i, 1) = 0$  then
8:     Add rule  $(f_i \in V_i \rightarrow 0)$  to  $R$ 
9:   end if
10:  end if
11: end for
12: for all  $i, j \in T$  do
13:   if  $\text{Count}(f_i \in V_i, f_j \in V_j, 1) = 0$  and  $\text{Count}(f_i \in V_i, f_j \in V_j, 0) > \text{threshold}$ 
   then
14:     Add rule  $(f_i \in V_i \wedge f_j \in V_j \rightarrow 0)$  to  $R$ 
15:   end if
16: end for

```

A few things to note:

- For the sake of legibility, we implicitly assume  $V_i, V_j$  to be the ‘right’ bins. In reality, membership to both  $V_i$  and  $\overline{V_i}$  are checked.
- Because  $P(f_i \in V_i | 0) = P(f_i \in V_i, 0) / P(0)$  and  $P(0)$  is a constant for a given training set, it is more efficient to work with joint probabilities.
- Because  $\text{Count}(f_i \in V_i, f_j \in V_j, 0) \leq \min(\text{Count}(f_i \in V_i, 0), \text{Count}(f_j \in V_j, 0))$ , we can already eliminate many combinations of feature dimensions to consider; this is the function of the set  $T$ . In our experiments, this reduces the number of combinations to check from 3.7 million to 30,000 and keeps the above algorithm tractable.

Furthermore, this algorithm gives us a tool to select negative examples in a semi-supervised fashion as well. The basic selection adapted from MIL relies on the augmented probability of having a positive label, given that the relation exists in the labelled data. As argued before, a similar reasoning does not hold for negatives, rendering selecting for them infeasible. However, the rules extracted by the above algorithm can serve not only to select away very unlikely candidates for positive

labelling, as previously done. As these rules try to encode sufficient conditions for negativity, we can also employ them to distinguish a subset of all the other unlabelled data as being very likely negative. This offers us the opportunity to add both positive and negative points from our unannotated corpus, a technique not feasible in the basic framework.

## Experiments

### Subject and data

The Gene Regulation Network Task tries to accomplish detection of relations overarching a diverse set of molecular interactions. Specifically, six different types of relations are to be extracted: inhibition, activation, requirement, binding, transcription and regulation. The training and development set consists of 134 sentences, jointly encoding 230 interactions. On average this amounts to 38 examples per relation type. Considering the specialised language and grammar often used in scientific publications, the amount of training data seems rather sparse to learn a good general representation in such a complex output space.

As previously argued, this is the main motivation for including additional data for use in the methods described above. We therefore augment the dataset we have with all sentences from PubMed abstracts responding to the query for “*bacillus subtilis sporulation*”<sup>[1]</sup>. Beginning from the annotated datapoints, we add a sentence from those unannotated texts if it contains at least two entities that also occur in our annotated data as having one of the above relations. Without these entities, a sentence could indeed never encompass a candidate data point for a relation. As such, from the initial 14109 sentences, only 1933 are retained. In Table 1 we have shown the average amount of data points that effectively got added to the training set for each system.

### Evaluation

#### *The Slot Error Rate*

From the predictions, a network gets constructed with the entities as the nodes and the relations between them as arcs. This network is then used for measuring performance: it gets compared to the reference by means of the *Slot Error Rate* (*SER*). This measure is defined by [15] as:

$$SER = \frac{S + I + D}{N}$$

with:

- $S$  the number of substitutions, i.e. edges that are predicted, but with the wrong type;
- $I$  the number of insertions (false positives);
- $D$  the number of deletions (false negatives);
- $N$  the number of arcs in the reference network.

For the following analysis, we further define

---

<sup>[1]</sup> <http://www.ncbi.nlm.nih.gov/pubmed/?term=bacillus+subtilis+sporulation>, accessed on 16/08/2013

- $C$  the number of correctly predicted relations;
- $M$  the number of arcs in the prediction.

The main motivation of [15] in proposing this error measure is the observation that  $F_1$ , the often-used harmonic mean of precision and recall, can be seen to be:

$$F_1 = 1 - \frac{S + (D + I)/2}{(N + M)/2}.$$

This derivation leads to believe that substitutions get overweighted in the use of this scoring mechanism. While by no means questioning the usefulness of the separate components (precision and recall), the *SER* gets proposed as a more balanced way of combining them as a means to compare systems.

The devil is however in the details; or rather, the denominator. While it is true that  $S$  gets a bigger weight in the numerator, one has to account for the weighting of the different components in the denominator, since

$$\frac{N + M}{2} = C + S + \frac{D + I}{2},$$

where we use that  $N = D + C + S$  and  $M = I + D + S$ . A similar weight scheme can hence be seen in the denominator as well, softening the argumentation against it. With a similar derivation, one finds:

$$SER = \frac{N - C + I}{N} = 1 - recall + \frac{I}{N}.$$

This insight shows us that in attempting to lower the weight for  $S$ , this error rate has become completely independent of this factor altogether! Furthermore, the unboundedness of this measure can be fully attributed to the number of insertions. This can explain the prevalence of conservative systems that this task has received: as can be seen from the official results, all but one submission have a very low number of arcs in their prediction, mostly attributable to a low  $I$  figure.

#### *Error measures: uses for comparison and model optimisation*

By this analysis, we wish by no means to imply that the *SER* is a *bad* scoring mechanism per se. This kind of *word error rates* is widely used in several research branches, and with good reason. However, as the name somewhat shows, these are situations where a more or less fixed number of *slots* need to be ‘filled’, such as (speech) phoneme recognition or named entity recognition. In our notation, this would be equivalent to  $M \cong N$ . If this constraint is taken into account, one can show that  $SER \cong 1.5(1 - F_1)$ , which is exactly what [15] find in their comparative analysis of measures.

In different settings however, where the above approximation is not sure to hold, the choice of *SER* implies an additional degree of freedom, of which the consequences are not evident to grasp. In this more general case, *SER* is seen to overly reward precision in a great part of the result space. This can even occur at the cost of recall, as will be shown below.

We believe there is an interesting opportunity for further research and discussion on this matter. Interesting, more general analyses can be found in both [16] and



[17]. In the light of this study however, we mainly wish to highlight the inherent bias towards precision this design choice entails. As we are investigating methods of obtaining recall, this is certainly a factor to take into account.

Comparison of performance between different systems (*intersystem* performance) is not the only function of a measure. The same measures get generally used for *intrasystem* measurements as well: in the comparison of multiple incarnations of models, and more commonly, hyperparameter optimisation. In order to assess the behaviour of the latter under different performance measures, we consider an ideally automated setting of optimising, not unlike running a gradient descent/ascent algorithm. In contrast to the case of general convex optimisation however, there is no convergence to a unique optimum. Rather, we are limited by the boundary of our system’s performance, generally known as the *performance-recall curve*: the maximum precision that can be obtained for any required recall. Hence, we are driven by the measure’s gradient until that border is reached.

As we can see in Figure 1, the gradient field of *SER* shows some interesting behaviour. In a substantial region of the recall-precision space, there is an enormous push towards increasing precision. In the region of precision below 50%, this even happens at the cost of maintaining recall. As a result, a system optimised for this measure will generally show good performance, but has little focus on improving recall. For comparison, the analogous field for the  $F_1$  measure is shown in Figure 2, which displays a better balance between favouring recall or precision, based on which is most lacking.

As previously argued, there are use cases where an adequate amount of recall is called for. With this in mind, we point out that  $F_1$  is embedded in a larger family of *F-measures*:

$$F = \frac{PR}{(1 - \alpha)P + \alpha R};$$

and we obtain  $F_1$  for the case of  $\alpha = 0.5$ . This parameter  $\alpha$  can be a great tool for the system or task designer to designate the proportion of importance he wishes to place on the precision/recall trade-off. If precision is to be targeted, a value of  $\alpha > 0.5$  will accomplish this, without having gradients go ‘against the grain’ of increasing both basic measures.

#### *Aggregation of predictions and impact on scoring*

A final concern is the aggregational processing that occurs before calculating the performance measures. In a traditional machine learning setup, scores are calculated in a *local* scope; meaning, every predicted point is compared to a ground truth, and from the numbers extracted for correct predictions, substitutions, insertions and deletions, the necessary proportions are calculated.

In the GRN task [2], performance is measured in a *global* fashion, due to the processing on the solution set that takes place before calculating the score. This happens in two steps:

- From the predicted classifications a network is built. All scoring is done with respect to this, implying that multiple classifications of a same relation get collapsed into one.

- ‘Resolution of redundant arcs’: recall that the different types of relations are ordered into a taxonomy (Table 3). Before scoring, any relation between two entities that is less specific (i.e. higher up the tree) than another appearing in the set, is removed.

We can see that this procedure renders the precision-recall trade-off a lot more intricate than in a traditional machine learning setting. In a *local* scoring procedure, the number of true positives can never decrease by adding more predictions; this is the main logic behind Receiver Operating Characteristic (ROC) curves as monotonously non-decreasing functions. Analogously, in the recall-precision space, this ensures a non-increasing curve of attainable points. Furthermore, this curve spans the whole range of recall: a recall of 100% is always attainable with a precision of at least the ratio of positives in the test set, a worst case that corresponds with classifying all test points as positive (see [18] for a thorough analysis of this and a performance measure that ensues from this, the Area Under Precision Recall Curve (AUCPR)). These principles no longer hold when removing predictions prior to measuring; adding a more specific prediction to an existing true positive renders the latter as non-existent, and recall at the end of the precision-recall curve will be limited by the ratio of positives that have the most specific relation (the leaves of the hierarchical tree in Table 3). This dynamic stands orthogonal to research on performance measures in a hierarchical setting (as in [19]), which is pursuing less level-dependence in assessing predictions.

This demonstrates that attaining sufficient recall is a greater challenge than in a regular setting. Furthermore, by adding a layer of complexity, it convolutes multiple tools that are basic in systems engineering: error analysis, model selection and comparison. We therefore wish to advocate the addition of *local*, unprocessed evaluation figures in future instalments of this task.

## Results and discussion

Results for our experiments can be seen in Table 2. The basic method we propose is entry [BASIC] in this table. Even without any added noise-balancing measures, this system can already be seen to showcase more than a doubling in recall compared to the original submission results of [3]. In light of the previous discussion, this demonstrates a manifest improvement in this dimension.

Results for the probabilistic pre-selection approach we developed can be found in entries [PRE-SEL] and [PRE-SEL + NEG]. In the former, we only include (and filter) positives from our unlabelled set, in the fashion of our basic method. The latter also employs the found rules to further add negatives from the unannotated corpus. Both are able to display a further improvement in F1, while still maintaining a good recall-precision balance. Especially the application of the filter to add negatives ([PRE-SEL + NEG]) warrants a substantial rise in F1 score through an additional improvement in precision compared to the [PRE-SEL] model. From Table 1 we can see that this model filters out about half of the potential negatives, to only retain the most informative ones.

These results confirm the value of filtering the unlabelled data before presenting them to the learning algorithm. As this has been done here exclusively based on the limited amount of labelled data, leveraging additional knowledge in this step could generate even more significant gains.

Differential regularisation (entry [DIFF-REG]), a method traditionally suggested to handle additional noise in semi-supervised frameworks, also obtains a high recall for this test. This comes however at a severe cost to its precision, compared to our newly developed solutions. This demonstrates the idiosyncratic nature of our methods as applied to this particular task with respect to mainline MIL methods, and further validates their contribution compared to utilising standard approaches.

Deeper study is required on the impact of our methods, since the performance of a system greatly depends on e.g. the features it uses. It remains an open question what the impact is of these implementation choices, such as the feature representation used, data preprocessing, etc. in comparison to the higher-level model choice. We suspect that a more fine-grained encoding of sentence context could further attribute to the performance of any system in this field.

## Related Work

In information extraction and relation extraction in particular a major bottleneck is the lack of sufficient annotated examples. The manual labelling of enough training instances in order to build an accurate classifier is often prohibitively expensive. On the other hand, collecting a large quantity of unlabelled textual data is cheap. Thus, it is interesting to train the extraction system on a small annotated corpus and in some way improve the quality of the learned classification patterns by exploiting the unlabelled examples. This has led to bootstrapping, semi-supervised and even unsupervised learning techniques.

The oldest methods regard self-training and co-training, where a classifier is trained iteratively. In self-training, examples from the pool of unlabelled instances are chosen in the next training step to which the current classifier assigns labels with most certainty. In co-training, examples are chosen in the next training step to which two or more current classifiers that possibly use an independent feature set assign labels with most certainty [20]. Such a set-up promotes that the newly introduced training examples have similar patterns as the originally labelled examples, so no radical new patterns are learned at least not in the first steps of the iteration. This approach also does not offer an answer to the danger that the obtained classification function drifts away from the real classification boundary. In a variant scenario, a generative probabilistic classifier is used (i.e., probabilities are not estimated directly, rather they are estimated indirectly by invoking Bayes' rule, e.g., a naïve Bayes classification) for the training of the initial classifier based on the seed set of labelled examples. The Expectation Maximization (EM) algorithm is then used to train the classifier that learns both from the labelled and unlabelled examples [21], but the algorithm can easily get stuck in a local maximum.

In so-called open domain information extraction frequently occurring patterns that signal a relation between two entities are identified in a large set of unlabelled data [22, 23]. These techniques are not well suited for the extraction of relations in the biomedical domain, especially when the detection of infrequent relations is targeted.

The relation extraction models that we present in this paper are closest to the work of [12]. These authors find sentences in Web documents that contain two given entities. It is a priori known that these entities are involved in the sought

relation. The selected sentences contain positive as well as negative examples of the sought relation. The negative examples for training the classifier are sentences in Web documents that contain two given entities for whom it is known that the sought relation does not hold between them. To cope with the noise in the set of positive examples, differential regularization is used when training a SVM, as we do in this paper. Our experiments on texts from the biomedical literature show that differential regularization did not yield the best results for semi-supervised learning. We have proposed a semi-supervised model with probabilistic pre-selection of positive and negative examples from the pool of unlabelled examples that makes use of the knowledge in the labelled examples in a demonstrated effective way in order to select unlabelled examples. This model improves the results of state-of-the-art differential regulation techniques.

## Conclusions and future work

We have explored the addition of unlabelled data to increase the recall of our system. However, the noisy nature of this data tends to affect precision negatively. We have designed a pipeline to autonomously counterbalance this effect, based on no additional external knowledge. A promising extension of this method would be to include specialised external knowledge, either injected directly into the feature representation, or in the process of attributing labels to unannotated data. This could prove to be a powerful technique in attaining a more precise overall system. Another interesting approach could be to construct a more extensive pipeline, using one of the more precision-bearing techniques to improve upon our proposed system.

Promising methods in general information extraction make use of language models (e.g., probabilistic models of word distribution) trained on huge amounts of unlabelled examples in order to find valuable replacements of words in the relation patterns or to identify valuable correlated word features used in the classification ([10, 24, 25]). Recent work in biomedical event extraction already touches upon such ideas ([26]). This is a path we intend to further explore in future work.

Another particularly interesting approach is showcased by [27], training a classifier jointly on both labelled and unlabelled data. A promising direction could be to apply similar methods to specialised language corpora, such as the biomedical texts explored in the BioNLP tasks.

We argue for the importance of recall in any information extraction task, to serve as a driving force for automated knowledge collection. This study contributes to gaining a deeper insight in the different factors at play in the 2013 BioNLP GRN task with respect to measuring performance, and the interplay of precision and recall in particular. We hope this will spark further discussion and analysis of both task organisation and submitted systems, thus helping this Shared Task in driving forward the field of biomedical IE.

## Abbreviations

*GRN*: Gene Regulation Network; *IE*: Information Extraction; *MIL*: Multiple-Instance Learning; *NLP*: Natural Language Processing; *SER*: Slot Error Rate; *SVM*: Support Vector Machine;

- *S* the number of substitutions, i.e. edges that are predicted, but with the wrong type;
- *I* the number of insertions (false positives);
- *D* the number of deletions (false negatives);
- *C* the number of correctly predicted edges;
- *N* the number of arcs in the reference network;
- *M* the number of arcs in the prediction.

### Competing interests

The authors declare that they have no competing interests.

### Author's contributions

TP is the lead writer and conducted all data processing and analyses, MFM oversaw the study, helped write the manuscript and provided overall guidance. All authors read and approved the final version of the manuscript.

### Acknowledgements

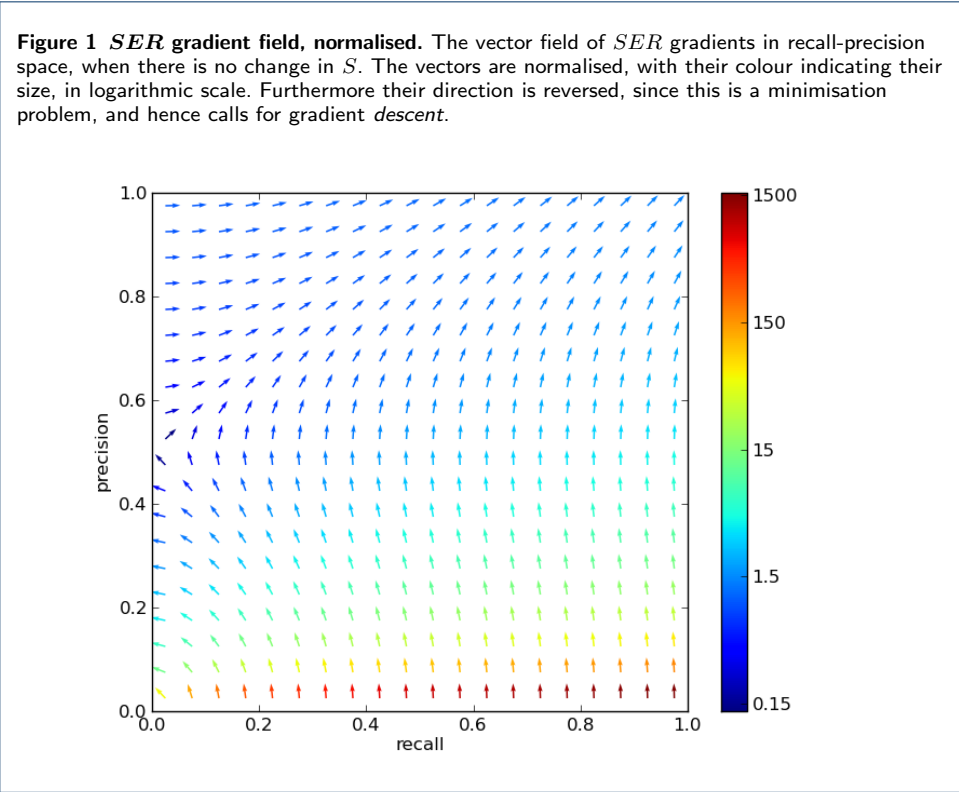
The authors would like to thank the Research Foundation Flanders (FWO) for funding this research (grant G.0356.12).

### References

- Nédellec, C., Bossy, R., Kim, J.-D., Kim, J.-J., Ohta, T., Pyysalo, S., Zweigenbaum, P.: Overview of bioNLP shared task 2013. In: *Proceedings of the BioNLP Shared Task 2013 Workshop*, pp. 1–7. Association for Computational Linguistics, Sofia, Bulgaria (2013)
- Bossy, R., Bessières, P., Nédellec, C.: BioNLP shared task 2013 - an overview of the genic regulation network task. In: *Proceedings of the BioNLP Shared Task 2013 Workshop*, pp. 153–160. Association for Computational Linguistics, Sofia, Bulgaria (2013)
- Provoost, T., Moens, M.-F.: Detecting relations in the gene regulation network. In: *Proceedings of the BioNLP Shared Task 2013 Workshop*, pp. 135–138. Association for Computational Linguistics, Sofia, Bulgaria (2013)
- Vapnik, V.N.: *The Nature of Statistical Learning Theory*. Springer, New York, NY, USA (1995)
- Aizerman, M.A., Braverman, E.M., Rozonoer, L.I.: Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control* **25**, 821–837 (1964)
- Schölkopf, B., Smola, A.J.: *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Adaptive computation and machine learning*. MIT Press, Cambridge, MA, USA (2002)
- Chang, C.-C., Lin, C.-J.: LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* **2**, 27–12727 (2011)
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011)
- Yang, L., Jin, R., Sukthankar, R.: Semi-supervised learning with weakly-related unlabeled data: Towards better text categorization. In: *NIPS*, pp. 1857–1864 (2008)
- Huang, F., Ahuja, A., Downey, D., Yang, Y., Guo, Y., Yates, A.: Learning representations for weakly supervised natural language processing tasks. *Computational Linguistics*, 85–120 (2013)
- Amores, J.: Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence* **201**(0), 81–105 (2013)
- Bunescu, R.C., Mooney, R.J.: Learning to extract relations from the web using minimal supervision. In: *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL'07)*, Prague, Czech Republic (2007)
- Cortes, C., Vapnik, V.: Support-vector networks. *Machine Learning* **20**(3), 273–297 (1995)
- Veropoulos, K., Campbell, C., Cristianini, N.: Controlling the sensitivity of support vector machines. In: *Proceedings of the International Joint Conference on AI*, pp. 55–60 (1999)
- Makhoul, J., Kubala, F., Schwartz, R., Weischedel, R.: Performance measures for information extraction. In: *Proceedings of DARPA Broadcast News Workshop*, pp. 249–252 (1999)
- Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. *Information Processing and Management* **45**(4), 427–437 (2009)
- Powers, D.M.W.: Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies* **2**(1), 37–63 (2011)
- Boyd, K., Costa, V.S., Davis, J., Page, D.: Unachievable region in precision-recall space and its effect on empirical evaluation. In: *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*. ICML '12, pp. 639–646. Omnipress, New York, NY, USA (2012)
- Silla, C.N. Jr., Freitas, A.A.: A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery* **22**(1-2), 31–72 (2011)
- Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: *Proceedings of the Eleventh Annual Conference on Computational Learning Theory. COLT' 98*, pp. 92–100. ACM, New York, NY, USA (1998)
- McCallum, A., Nigam, K., Rennie, J., Seymore, K.: A machine learning approach to building domain-specific search engines. In: *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2. IJCAI'99*, pp. 662–667. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1999). <http://dl.acm.org/citation.cfm?id=1624312.1624313>
- Etzioni, O., Fader, A., Christensen, J., Soderland, S., Mausam, M.: Open information extraction: The second generation. In: *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume One. IJCAI'11*, pp. 3–10 (2011)
- Mitchell, T.M.: Never ending learning. In: *Proceedings of the European Conference on Artificial Intelligence*, p. 5 (2012)
- Deschacht, K., De Belder, J., Moens, M.-F.: The latent words language model. *Computer Speech and Language* **26**(5), 384–409 (2012)
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: *Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 26*, pp. 3111–3119 (2013)
- Wang, J., Xu, Q., Lin, H., Yang, Z., Li, Y.: Semi-supervised method for biomedical event extraction. *Proteome Science* **11**(Suppl 1):S17 (7 November 2013) (2013)

27. Riedel, S., Yao, L., Marlin, B.M., McCallum, A.: Relation extraction with matrix factorization and universal schemas. In: Joint Human Language Technology Conference/Annual Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL '13) (2013)

Figures



Tables

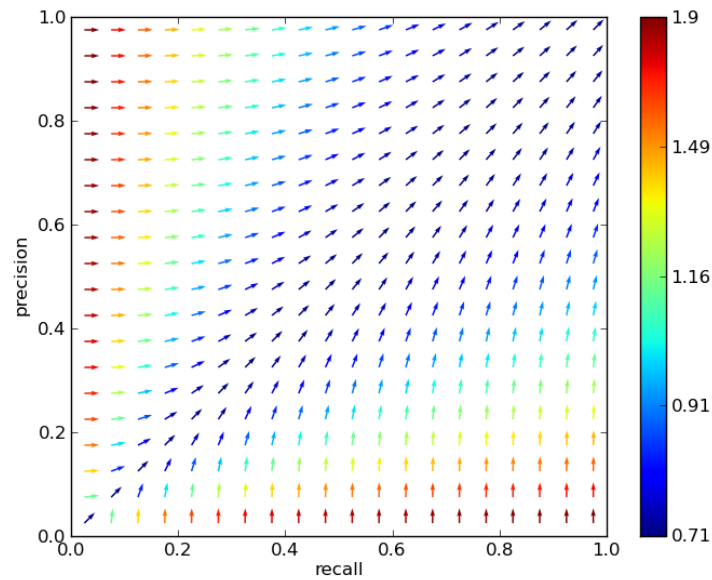
**Table 1** Average number of data points from unannotated corpus used in systems. Also mentioned is the percentage of the total candidate pool this is.

System	positives	negatives
[BASIC]	679	0
[PRE-SEL] and [DIFF_REG]	425 (= 62.6 %)	0
[PRE-SEL + NEG]	425	1417 (= 52.0 %)

**Table 2** Comparative table of results for our different systems. All results are on the official test set. (*S* = substitutions, *I* = insertions, *D* = deletions, *C* = correct predictions, *M* = number of predictions, *N* = 88 = number in reference).

System	S	D	I	C	M	Recall	Prec.	F1	SER
Original submission of [3]	15	53	5	20	40	22.7	50.0	31.3	0.830
[BASIC]	28	18	100	42	170	47.7	24.7	32.6	1.659
[DIFF_REG]	30	12	204	46	280	52.3	16.4	25.0	2.795
[PRE-SEL]	28	20	77	40	145	45.5	27.6	34.3	1.420
[PRE-SEL + NEG]	17	34	41	37	95	42.0	38.9	40.4	1.045

**Figure 2**  $F_1$  gradient field, normalised. The vector field of  $F_1$  gradients in recall-precision space. The vectors are normalised, with their colour indicating their size, in logarithmic scale. Of note is the scale difference with Figure 1.



**Table 3** Hierarchy imposed on the output types of the GRN task.

- Regulation
  - ↔ Binding
    - ↔ Transcription
  - ↔ Activation
    - ↔ Requirement
  - ↔ Inhibition